

Title: Method and Apparatus for an E-mail Affiliate Program

Inventor: John Ferber

1 RELATED APPLICATIONS

2 This application claims the benefit of U.S Provisional Application No. 60/166,690, titled
3 "E-mail Affiliate Program and Process," filed November 20, 1999.

4
5 FIELD OF THE INVENTION

6 This invention is drawn to an e-mail affiliate program and process for marketing
7 purposes. It includes software for the implementation of a system to produce e-mail marketing
8 lists. A preferred embodiment for these lists is to supply subscribers for Internet publications
9 commonly known as "ezines".

10
11 BACKGROUND OF THE INVENTION

12 In recent years, the exponential growth of the network of computer networks known as
13 the Internet has also lead to enormous growth in the area of "on-line" advertising. One popular
14 channel of on-line advertising has been e-mail.

15 Typically, entities have collected e-mail addresses from various sources, such as by
16 manually collecting or using spiders or bots to collect e-mail addresses from news groups or
17 auction sites on the Internet. They then broadcast or "spam" an identical, unsolicited marketing
18 message to their collected list. Needless to say, these unsolicited messages annoy most recipients
19 and result in very low response rates. Additionally, many recipients regard the "spam" as an
20 invasion of their privacy. What is needed is a system that sends marketing messages only to those

1 who "opt-in" and increases both the number of participants and/or the number of responses by
2 offering incentives.

4 BRIEF SUMMARY OF THE INVENTION

5 The present invention is drawn to software that allows an enterprise to offer cash, prizes,
6 or incentives to webmasters or consumers in exchange for their assistance in (i) accumulating
7 subscribers or e-mail addresses and/or (ii) getting subscribers to view advertisements or to refer
8 associates to the program.

9 The subscribers (typically subscribed via their e-mail address) may subscribe for content-
10 based subscriptions/newsletters/alerts, marketing message only-based subscriptions/mailings,
11 and incentive-based subscriptions where the subscriber may receive incentives, such as payment
12 in cash, program points toward prizes, discounts on products or opportunities to win
13 sweepstakes, in exchange for responding to future e-mails and other marketing messages or
14 content delivered through internet-enabled channels.

15 The present invention includes:

- 16 a) a sign-up means that allows webmasters or consumers to enter their information,
17 including e-mail address, and the right to market to it, online through a web browser
18 so that they can participate into the program;
- 19 b) a set of instructions and computer coding into which the webmaster or participant
20 must use as a method of which to promote the program , including prompting the user
21 for an action, like entering information or clicking somewhere, before they are
22 eligible;
- 23 c) a statistical interface which provides the webmaster or participant with a way to see

how many subscribers and/or referrals they have generated and or what revenue/prizes/incentives they have accumulated to date (either instantly directly after the transaction, or at any time as a cumulative sum of all chance possibilities that have occurred in the past); and

d) a software application which allows an administrator of the program to

1) view accounts of participants,

2) view number of participants,

3) edit information on participants,

4) keep track of subscriber and webmaster referral information, and

5) send e-mails to the subscribers and/or the participants.

The software allows a consumer to sign-up to subscribe for delivery of information or content such as ezines, newsletters, alerts or marketing messages with each such delivery to be sent to the subscriber's e-mail address. Consumers may subscribe for more than one offering (i.e., co-registration).

The e-mail delivery system of the present invention includes a system that consists of software and hardware allowing the owner of a mailing list the ability to e-mail out marketing messages to their subscribers and enable them to track the responses of those messages.

Tracking of responses includes:

- a) whether the subscriber received the e-mail, opened the e-mail, read the e-mail, how long they read the e-mail, did they perform any action inside of the e-mail such as fill in information, click on a hyperlink;
- b) the geographic location of the subscriber(s), area code, domain name, ISP, sex, marital status, occupation, etc... this reporting can be in the form of individually

1 reported data or group reported data;

- 2 c) the ability to track subscribers over time in regards to all of the e-mails they have
3 received, the ads they have received, the ads they have responded to, characteristics
4 about that subscriber, including, but not limited to, observed behavior, demographic
5 and psychographic data that is available on the subscriber;
- 6 d) the ability to track on a timeline when any of the actions, such as opening, clicking,
7 etc., occurred for the population sample mailed to; and
- 8 e) the ability to send different messages to sample groups with the same info, decision
9 which message is most effective, and send more of the better-performing message to
10 the rest of the population with the same or similar demographics/details.

11 It is an object of the present invention to provide software that allows an enterprise to
12 offer cash, prizes, or incentives to webmasters or consumers in exchange for their assistance in
13 (i) accumulating subscribers or e-mail addresses and/or (ii) getting subscribers to view
14 advertisements or to refer associates to the program.

15 It is an object of the invention to provide the ability to reliably deliver marketing
16 messages to millions of recipients on a periodic or demand basis.

17 It is a further object of the invention to store e-mail marketing information directly in a
18 database instead of flat files or other file formats.

19 It is another object of the invention to import existing participant e-mail marketing
20 information from flat files or other file formats into a new database.

21 It is another object of the invention to keep e-mail marketing statistics about each e-mail
22 subscriber, including: lists subscribed (including start and end date), total number of e-mails sent
23 to this subscriber, which ads e-mailed to this subscriber, and which ads clicked by this

1 subscription/unsubscription per day per particular content, and bandwidth usage.

2 It is another object of the invention to remove bogus/undeliverable addresses in an e-mail
3 marketing system.

4 It is another object of the invention to provide an e-mail marketing program with real-
5 time or batch subscribe and delete facilities via a web or an e-mail interface.

6 It is an object of the invention to provide the ability to show ads on “public” web
7 interfaces to the content delivery system in an e-mail marketing program (i.e., when subscribers
8 confirm their subscription, show an ad on that page).

9 It is an object of the invention to provide documentation on how each component of an e-
10 mail marketing system works.

11 It is an object of the invention to provide the ability to test an e-mail marketing system.

12 It is an object of the invention to provide a web or an e-mail interface to allow e-mail
13 marketing subscribers to retrieve ezine back issues with new ads.

14 It is an object of the invention to provide personalized content and ads for individual
15 recipients, specifically selecting content based upon attributes in a subscriber database (e.g.,
16 gender, age, chosen topics, etc.).

17 It is an object of the invention to provide an HTML sniffer to determine if a given e-mail
18 marketing subscriber can receive HTML e-mail.

19 It is an object of the invention to provide the ability to add/edit/modify ads stored in a
20 database via web interface.

21 It is an object of the invention to provide the ability to detect fraud in an e-mail marketing
22 program.

23 It is an object of the invention to provide the ability to monitor e-mail marketing system

1 CPU/memory/disk/throughput usage on an hourly/daily/monthly or other assessment period
2 basis.

3 It is an object of the invention to provide the ability to gather info about qmail usage via
4 qmailanalog package.

5 It is an object of the invention to integrate ad responses in an e-mail marketing system
6 with an advertising server, including the ability to track which ads a subscriber responded to.

7 It is an object of the invention to provide the ability to send cookies to subscribers when
8 they confirm their subscription to an e-mail marketing program and link these up with advertiser
9 server cookies.

10 It is an object of the invention to provide the ability to construct new (temporary) "sub-
11 lists" from any set of e-mail marketing subscribers in the database via a web interface.

12 It is an object of the invention to provide the ability for content creators to automatically
13 include ads in an issue without manually inserting lengthy code.

14 It is an object of the invention to keep e-mail marketing statistics about each ad e-mailed,
15 including total times each ad e-mailed, how many times each ad e-mailed per ezine, total times
16 each ad responded to, and total unique responses for each ad.

18 BRIEF DESCRIPTION OF THE DRAWINGS

19 **Figure 1** illustrates a basic network architecture for practicing the present invention.

20 **Figure 2** illustrates a basic flow diagram for practicing the present invention.

22 DETAILED DESCRIPTION OF THE INVENTION

23 As used herein, the term "webmaster" refers to any entity, including an individual or

from other tables, and so have unique primary (i.e., not composite) keys.

Confirmation Table

Each row of this table describes one subscriber's confirmation information. Each row need only persist until a subscriber has confirmed their subscription, and can then be deleted.

Columns:

- email (primary key) // required, varchar(80), references email field in user table
- ezine_code_list // required, varchar(60), space-delimited list of ezines initially signed up for, ignored once they have confirmed their signup - look in ezine_subscriber table instead.
- referring_account_number // required, varchar(80), references account_number in webmaster table for initial signup (before confirmation). Ignored after subscriber confirms - look in ezine_subscriber table instead.

User Table

Each row of this table describes information common to any participant (either subscriber or webmaster)

Columns:

Info for all participants (webmasters or subscribers)

- email (primary key) // required, varchar(80)
- first_name // required, varchar(30)
- last_name // required, varchar(30)
- zip_code // required, char(10)
- country // required, varchar(30), range is list of provided countries
- signup_datetime // required, datetime, this is the time when the subscriber is mailed their initial signup letter
- IP_address // required, char(15)
- is_valid_email // required -- is this a valid email

- address, char(1), Y|N
- admin_comments // optional, text, any comments that we want to add about this user. This is not information the user provides.
- is_active // optional, char(1), Y|N is this user "active," i.e., can they receive ezines and be paid
- last_modified_timestamp // timestamp, not null
- num_bounced_deliveries // int, number of delivery attempts to this user which have bounced

Subscriber Table

Each row of this table describes one subscriber.

Columns:

- email (foreign key) // required, varchar(80), references email field in user table
- confirmation_string // required for S, char(64), md5(email)+ datetime + process ID
- cookie_info // required for S, varchar(100)
- is_HTML_reader // required, char(1), Y|N -- can this subscriber receive HTML-ified ezines

All these are optional info for subscribers. Different ones will be displayed to different subscribers when they sign up, based on the referring webmaster.

- gender // char(6), Male or Female
- age // char(10), ranges: <13, 13-17, 18-21, 22-24, 25-29, 30-35, 36-40, 41-49, 50-59, 60-65, 65+
- education_level // varchar(40), ranges: "some high school," "high school graduate," "some college," "vocational/technical training," "college graduate," "graduate school," "doctorate"
- marital_status // char(7), range is married|single
- num_children // char(2), range is 0, 1, 2, 3, 4, 5+
- occupation // varchar(40), range: "professional," "administrative/clerical," "management," "designer/artist," "Internet professional," "educator," "marketing/sales," "engineer," "writer," "homemaker," "military service," "consultant," "legal field,"

1 "student," "retired," "other"
 2 • income // varchar(10), range: <25K, 25-49K, 50-74K, 75-
 3 99K, 100-149K, 150-199K, 200K+
 4 • primary_computer_use // char(40), range is "home,"
 5 "business," "home office," "school," "government
 6 facility"
 7 • interests // text, range is a comma-delimited list of
 8 items that appear as checkboxes on the signup page:
 9 sports, travel, entertainment/humor, dining, investments,
 10 gaming, computing, autos, men's issues, women's issues,
 11 health issues, trivia, astrology
 12 • items_purchased // varchar(255), any items the
 13 subscriber has purchased on the net
 14 • like_to_purchase // varchar(255), any items the
 15 subscriber would like to purchase on the net
 16 • plan_to_purchase // varchar(40), any items the subscriber
 17 is planning to purchase in the next year, range:
 18 "vehicle," "boat," "house," "computer equipment,"
 19 "vacation package," "stereo equipment," "VCR"
 20 • net_access_frequency // varchar(40), "every day," "once
 21 or twice a week," "two or three times a month," "once a
 22 month or less"
 23 • wants_to_receive_email // char(1), Y|N -- does this
 24 subscriber want to receive email about items of interest
 25 • last_modified_timestamp // timestamp, not null
 26 • email_md5 // varchar(64), not null, this is the md5 hash
 27 value of the email address
 28

Webmaster Table

Each row of this table describes one webmaster.

Columns:

- email (foreign key) // required, varchar(80), references email field in user table
- account_name // required, varchar(25), must be >= 6 chars
- account_number (primary key, auto_increment) // required, integer
- password // required, varchar(25), must be >= 6 chars
- address_line_1 // required, varchar(30)
- address_line_2 // optional, varchar(30)

- city // required, varchar(25)
- state // required, char(2), range is a list of state abbreviations, or N/A to indicate outside the US
- phone_number // required, varchar(20)
- payee_on_check // required, varchar(50)
- minimum_check_value // required, decimal(7,2)
- tax_ID_or_SSN // required, varchar(20)
- referral_rate // required -- for user referrals, decimal(4,2), default ".10"
- second_tier_rate // required -- for second-tier referrals, decimal(4,2), default ".04"
- referring_account_number // optional, varchar(80), the user who referred this one, references account_number field in webmaster table
- referring_URL // optional, varchar(80), website through which this webmaster was referred
- last_modified_timestamp // timestamp, not null

Sign-up Field Table

Each row of this table describes one possible field of data we want to collect for users.

Columns:

- field_name (primary key) // required, varchar(30), should exactly match one of the optional subscriber fields in the user table
- display_name // required - what the user sees on the web page, varchar(255)
- data_type // required, varchar(12), range is INT, STRING, DATE, etc.
- HTML_input_type // required -- type of control to show for this field on web page, varchar(12), range is checkbox, textfield, text, radio button, etc.
- value_range // optional, text, comma-delimited range if provided, else free-form text if not provided

Ezine Description Table

Each row of this table describes one ezine.

Columns:

- ezine_code (primary key) // required, char(2), two-character ezine code
- ezine_name // required, varchar(30), ezine name
- ezine_URL // required, varchar(80), ezine location
- short_description // required, varchar(80), description in a few words
- long_description // required, varchar(255), several sentence description
- thank_you_text // required, varchar(255), for signup
- is_active // required, char(1), Y|N -- is this an active ezine
- creation_datetime // datetime, when this ezine was created

Ad Table

Each row of this table describes one ad that can be inserted into an ezine issue

Columns:

- banner_id (primary key) // required, varchar(30)
- banner_text // required, TEXT
- banner_URL // required, varchar(80)
- num_clicks // required, BIGINT DEFAULT 0
- num_impressions // required, BIGINT DEFAULT 0
- is_active // required, char(1)
- entry_datetime // required, datetime

Advertiser Table

Each row of this table describes one advertiser

Columns:

- system_account (primary key) // varchar (50) not null
- advertiser_name (unique) // varchar (50), not null

Ad Campaign Table

1 Each row of this table describes one advertiser campaign

2 Columns:

- 3 • campaign_id (primary key) // varchar(30) not null
- 4 • system_account (unique) // varchar (50) not null,
- 5 references system_account in advertiser table
- 6 • banner_id // varchar(30) not null, references banner_id in
- 7 ads table
- 8 • num_clicks // bigint default 0
- 9 • num_impressions // bigint default 0
- 10 • start_datetime // datetime
- 11 • end_datetime // datetime
- 12 • comments // text
- 13 • gross_payment // decimal(5,2)
- 14 • percent_done // int(3)
- 15 • target_percent // int(3)
- 16 • unit_difference // bigint
- 17 • effective_CPM // bigint

18
19 The following tables are derived from information in the basic tables above. They also
20 contain some information unique to themselves. Note that these tables could have duplicate ID
21 keys, but will have unique composite keys.

22 23 *Website Table*

24 Each row of this table describes a webmaster and a website (since a webmaster might
25 own several different sites).

26 Columns:

- 27 • site_URL (primary key) //required, varchar(80)
- 28 • site_name // required, varchar(80)
- 29 • site_description // required, varchar(255)
- 30 • account_number (foreign key) // required, varchar(80),
- 31 references account_number field in webmaster table

Website Sign-up Field Table

Each row of this table describes a sign-up field to be displayed to new subscribers when they are referred by the specified website.

Columns:

- site_URL (foreign key) // required, varchar(80), references site_URL field in website table
- field_name (foreign key) // required, varchar(30), references field_name in signup field table
- is_required // required -- is this subscriber required to fill in this field when they signup from this site, char(1), Y|N

Website Hosted Ezines Table

Each row of this table describes one ezine that can be hosted by a given website when the website is being used by a webmaster to generate subscriptions.

Columns:

- ezine_code (foreign key) // required, char(2), references ezine_code in ezine table;
- site_URL (foreign key) // required, varchar(80), references site_URL in signup field table

User Payout Table

Each row is the payout for a given participant on a given date if the participant is to be paid for the subscription.

Columns:

- account_number (foreign_key) // required, varchar(80), references account_number in webmaster table
- amount_paid // decimal(14,5), not null
- date_paid // required, date, when paid

Ezine Subscribers Table

Each row of this table describes one subscriber to a given ezine

Columns:

- email (foreign key) // required, varchar(80), references email in user table
- ezine_code (foreign key) // required, char(2), references ezine_code in ezine table
- signup_datetime // required, datetime, when user signed up to receive this ezine
- last_modified_timestamp // timestamp, this is the last time that the user modified their subscription to this ezine. Initially this is the same as signup_datetime.
- is_active // required, char(1), Y|N-- is active recipient of this ezine
- referring_account_number // optional -- the first (and only) user who referred this subscriber to this ezine, varchar(80), references account_number key in webmaster table
- referring_URL // optional, varchar(80), the website through which the user subscribed to the ezine

Ezine Issue Table

Each row of this table describes one mailing or issue of any ezine

Columns:

- issue_id (primary key) // required, varchar(30)
- ezine_code (foreign_key) // required, char(2), references ezine_code in ezine table
- last_modified_timestamp // required, timestamp
- req_start_datetime // datetime, when delivery of this issue was requested to begin
- actual_start_datetime // datetime, when delivery of this issue actually began
- end_queue_datetime // datetime, when this issue was fully queued
- end_delivery_datetime // datetime, when issue's delivery completed
- status // required, char (1), status of this issue's

- delivery D|S|P|I (disabled, sent, pending, in progress)
- num_recipients // bigint, not null

Ezine Ad Stats Table

Each row of this table describes one ad's impression and click statistics for a given ezine or individual mailing of an ezine.

Columns:

- banner_id (foreign key) // required, varchar(30), references banner_id in ad table
- num_clicks // required, bigint, how many times this ad has been clicked from this ezine
- position_in_ezine // required, int(3), position of ad in ezine
- issue_id (foreign_key) // required, varchar(30) references issue_id in ezine_issue table

Subscriber Click Stats Table

Each row of this table describes one ad that a subscriber has clicked on

Columns:

- email_md5 (foreign_key) // required, varchar(64), references email_md5 in subscriber table
- banner_id (foreign_key) // required, varchar(30), references banner_id in ad table
- issue_id (foreign_key) // required, varchar(30), references issue_id in ezine_issue table

The following tables are for ezines administration purposes

Admin User table

Columns:

1 The software of the invention also provides various web interface tools for the ezine system,
2 including those used by ezine administrators, webmasters, subscribers, including:

3 • Ezine Content Creation

4 This allows a list creator/moderator to submit content for a list.

5 • Distribution Administration

6 Allows ezine administrator to indicate start time for distribution of one or more ezines.

7 Also allows administrator to cancel one or more distributions in progress. Administrator
8 should be able to add or remove subscribers or webmasters, as needed.

9 • Stats Viewer

10 Allows ad server personnel to view/extract database stats about ads, lists, webmasters,
11 individual ezines, or subscribers.

12 • Subscriber Sign-up

13 Allows subscribers to sign up for one or more ezines or other information they want. At
14 this point, the user enters their e-mail address and selects the ezines or other information
15 they want. A confirmation number is e-mailed to them, which they must use to confirm
16 their sign-up. If possible, a cookie is put on their device at this point.

17 • Subscriber Confirmation

18 This is where the user confirms their sign-up for the selected ezines. They are optionally
19 required to fill out a survey. Once completed, they are added to the mailing list for the
20 desired ezines or other information.

21 • Subscriber Ezine Change

22 Allows subscribers to sign-up for new ezines or cancel subscription to ezines they're
23 already receiving.

- Webmaster Sign-up

Allows webmasters to sign up, and indicate a referring (second-tier) webmaster. The webmaster is required to fill out a survey of information.

- Webmaster Stats/Admin

Allows webmasters to view the stats for the number of subscribers and webmasters they've referred to the program (and the payout they'll receive). They can also change their account info and get the required HTML code to place on their webpages or get other information needed to refer subscribers.

- Conversion utility

Imports existing flat files into a database (i.e., MySQL or Oracle). These are a series of perl or other language scripts that process the various flat files, and write a handful of new flat files which contain the combined data. These new flat files can then be imported into the database via the LOAD or other command.

- Bounce remover

Removes undeliverable addresses from the database. To facilitate this, the present invention adds its own header to each e-mail sent: X-Sent-To: address@domain.com.

An example of the webmaster coding which can be provided as part of this invention is disclosed in the following JavaScript:

```
<script language="javascript">
function thankyou_ne() {
    var w =
window.open("", "thankyou", "scrollbars=yes, resizable=no, toolbar=n
o, directories=no, status=no, menubar=no, location=no, screnX=200, scr
```

```

1  eenY=200,width=640,height=500");
2      return true;
3  }
4  </script>
5

```

6 This is how the <form> tag is defined:

```

7  <form target="thankyou"
8  action="http://webserver1.teknosurf.com/cgi-bin/subscribe.cgi"
9  method="post" onsubmit="return (
10 (this.email.value.indexOf(&quot;@&quot;)&gt;0 &amp;&amp;
11 this.email.value.indexOf(&quot;.&quot;)&gt;0) ? thankyou_ne() :
12 (alert(&quot;Please enter a valid email address.&quot;)) ||
13 false) )">
14

```

15 The following line is added to the form:

```

16 <input type="hidden" name="code_num" value="1">
17

```

18 In operating the present invention, it is preferable to use a webmaster's account_number
19 instead of account_name in referring_account field. This requires changes to the import script.
20 Also, if user logs in via account_name, the system can get the corresponding number and pass
21 that along to the webmaster-*.cgi scripts.

22 **Figure 1** illustrates a basic network architecture for practicing the present invention,
23 wherein an ad server **110** supplies the needed software to webmaster servers **120** and subscriber
24 devices **130**, either directly or indirectly (i.e., via another webmaster server or subscriber device
25 which already has downloaded the software) over internet **100**.

26 In a preferred embodiment, the ad server **110** provides both an ad server database **112** and
27 a referral/payout database **114**. The system also can include a separate list server **140**, that
28 maintains its own subscriber list database **142**, and a separate content server **150**.

29 **Figure 2** illustrates a basic flow diagram for practicing the present invention. An initial

1 step **200** in the process is for webmasters to sign-up and download the software needed to
2 practice the invention from an enabling entity such as an ad server. This sign-up can be done in
3 any suitable manner, but is preferably accomplished through use of a web-based sign-up form or
4 page, wherein the webmaster provides the input required for the user table and the webmaster
5 table, as described above. This input can also include any referring webmaster information.

6 Consumers, likewise at **220**, sign-up and download the software they need to practice the
7 invention from an enabling entity such as an ad server or webmaster server. Again, this sign-up
8 can be done in any suitable manner, but is preferably accomplished through use of a web-based
9 sign-up form or page, wherein the consumer provides the input required for the confirmation
10 table, user table, and subscriber table, as described above. This input can also include any
11 referring webmaster or referring subscriber information.

12 As participants, the webmasters and subscribers have the necessary software/code to
13 refer, at **230**, other webmasters and subscribers and this information is tracked by the ad server,
14 at **250**.

15 The information or content that the subscriber has opted-in to receive is e-mailed to the
16 subscriber at **240**, and the subscriber responses are tracked at **250**.

17 The information collected is then used for purposes such as to calculate incentives and
18 payouts for the program participants, at **260**. Additionally, the ad server can analyze the collected
19 information by categories/fields, including but not limited to, website, ezine, ad campaign, ad, ad
20 position, subscriber (including demographics, etc.), time, etc. to improve the effectiveness of the
21 marketing response.

22 As disclosed above, the present invention makes it easy to collect, manage, and
23 communicate with consumers using e-mail and other forms of push-based messaging as the



1 primary vehicle for communication of marketing content. The present system makes it simple to
2 offer bounties or other types of compensation to any participant interested in referring new users
3 into the system as well as offer a multiple tier referral system so that those participants can refer
4 others as well. The present invention makes it possible to track the entire path of a
5 communication and all of the actions inside of it for modeling, reporting, auditing and other
6 analytical purposes. The system also simplifies the process for generating custom marketing
7 communications in large capacity. The present invention simplifies the process of enabling the
8 selection, insertion and delivery of marketing communications inside of the main communication
9 layer and enables an administrator to easily administer all aspects of the software from an
10 intuitive web based interface. As disclosed, it enables the systems to run self-sufficient and be
11 monitored for maximum performance. The present invention also enables the entire system to be
12 scaleable.